Indica varios SGBD y clasifícalos en función de su arquitectura

1. **Arquitectura Cliente-Servidor**

Esta es la arquitectura mas conocida.

Atendiendo a la filosofía cliente servidor, existe la programación por capas, en la que se aplican 3 niveles.

Aunque no siempre las tres capas están separadas físicamente es conveniente que lo estén lógicamente

**Capa de presentación**: Es la que permite al usuario interactuar con la base de datos y mostrarle la información que este necesita.

Se conoce como interfaz de usuario

**Capa de negocio**: En esta capa se encuentra toda la lógica de la aplicación, todo el software intermedio y será donde se relacionen todos los procesos de tratamiento de la información obtenida en la capa de datos y cuyos resultados se muestran al usuario mediante la capa de presentación.

**Capa de datos**: En esta capa se encuentran los datos.

En ella se encuentran los SGBD y reciben peticiones para acceder o escribir datos desde la capa de negocio.

Proporciona al servidor de aplicaciones los datos necesarios para poder procesar y generar el servicio que solicito el cliente en un principio.

Graphical user interface, application

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

1. **Arquitectura Monolítica**

Se asocia con los sistemas heredados

Son pilas de aplicaciones únicas que contienen todas las funciones dentro de cada aplicación. Tienen conexión directa tanto en la interacción de los servicios como en la manera que se desarrollan y se distribuyen.

Un solo cambio en el código de la aplicación implica volver a lanzarla por completo. Por eso las actualizaciones y las versiones nuevas suelen lanzarse cada 6 meses o cada año y no deben incluir características nuevas, sino solo el mantenimiento general.

Diagram

Description automatically generated

1. Arquitectura microservicios

No so solo un tipo de arquitectura sino también una forma de abordar la escritura del software.

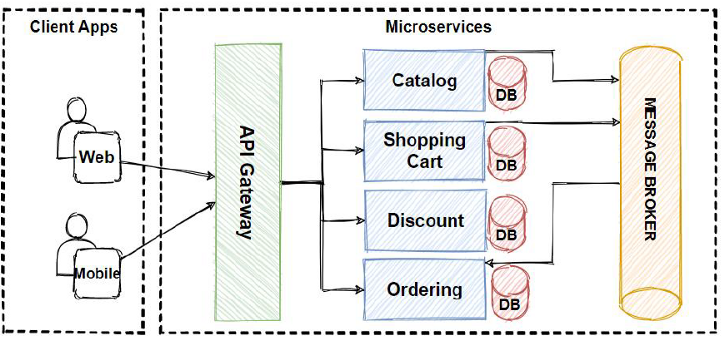
Con ellos, las aplicaciones se dividen en sus elementos mas pequeños, que son independientes entre si.

Cada uno de dichos elementos o procesos es un microservicio.

Esta arquitectura aporta beneficios tanto de estabilidad dinámica como de tolerancia a fallos.

Permiten desarrollar un software de calidad mas rápido ya que varios desarrolladores pueden trabajar en ellas al mismo tiempo

Junto con los equipos de DevOps y API, los microservicios en contenedores constituyen la base de las aplicaciones nativa de la nube.



1. **Arquitectura basada en eventos**

El procesamiento y la permanencia de los eventos son la estructura central de la solicitud.

Los eventos son aquellos sucesos o cambios significativos en el estado del hardware o el software de un sistema. Los eventos suceden con estímulos internos o externos.

Modelos de las arquitecturas basadas en eventos:

* Modelo de publicación y suscripción:

Es una infraestructura de mensajería que se basa en subscripciones a un flujo de eventos. Con este modelo, una vez se genera o publica el evento, este se envía a los subscriptores que necesitan estar informados al respecto.

* Modelo de flujo de eventos:

Con este modelo, los eventos se acriben en un registro. Los consumidores no se subscriben a un flujo de eventos, sino que pueden leerlo desde cualquiera de sus partes y unirse a el en cualquier momento.

* Procesamiento de flujos de eventos
* Procesamiento de eventos simple
* Procesamiento de eventos complejo

Diagram

Description automatically generated

1. **Arquitectura orientada al servicio**

Es un estilo consolidado de diseño de software que se asemeja estilo de la arquitectura de microservicios.

Cada uno de los servicios individuales se organiza en torno a un proceso empresarial especifico, todos cumple con un protocolo de comunicación (como SOAP, ActiveMQ o Apache Thrift) y se exhiben a través de la plataforma de un ESB (Enterprise Service Bus).

Diagram, engineering drawing

Description automatically generated

Al menos, indica una SGBD cuya arquitectura sea monolítica, otra de miscroservicios, basada en eventos y otra orientada al servicio.

**Arquitectura monolítica: Los Libros de Reactive Programing**

Diagram

Description automatically generated

**Arquitectura de microservicios:**

A red and white sign

Description automatically generated with low confidence

Arquitectura basada en eventos: una e-comerce

Diagram

Description automatically generated

**Arquitectura orientada al servicio:**

Ejemplo:

* Cisco adopted SOA to make sure its product ordering experience was consistent across all products and channels by exposing ordering processes as services that Cisco’s divisions, acquisitions, and business partners could incorporate into their web sites.

Source: https://www.ibm.com/uk-en/cloud/learn/soa